realtimepublishers.com™

The Definitive Guide™ To

Windows Desktop Administration

Bob Kelly

Chapter 2: OS Deployment 23
The Workstation Baseline 23
Benefits 23
Increased Reliability 23
Increased Deployment Speed 23
Ease of Troubleshooting 24
Drawbacks 24
Baseline Components 24
What to Leave In 25
What to Leave Out 25
Common Pitfalls to Avoid 25
MSI Source Resiliency 25
Non-Essential Data 26
Classified or Company Proprietary Information 26
Globally Unique Identifiers 27
Profiles 27
Security Identification Numbers 28
Initial Build Size 30
Staging a Deployment 32
Benefits 32
Less Time Spent at the Users' Desks 32
Controlled Network Environment 33
Early Identification of Failed Systems 33
Drawbacks 34
Location-Specific Actions 34
Slow Delivery in a Changing Environment 34
Deployment Methods 34
Manual Installation 34
Unattended Installations 36
Microsoft Solutions 37
Sysprep 37
Unattend.txt 39
OEM 40

Cmdlines.txt 40
Third-Party Solutions 40
Drive Imaging 41
Available Solutions 43
Which Is Right For You? 46
Drive Duplication 46
Available Solutions 47
OS Deployment Best Practices 47
Workstation Naming Conventions 47
System Build Information 49
Image Identification 49
Summary 49

Copyright Statement

Chapter 2: OS Deployment

In the last chapter, we touched on each of the primary areas of desktop administration. In the remaining chapters, we will drill down into each of these topics in much more detail. This chapter covers the first of these subjects, operating system (OS) deployment. The deployment of new systems is the foundation on which the stability of the workstation will be set, no matter what the makeup of that new system—a new computer, a new version of Windows, or simply a new configuration. A poorly built image, a bad script, or poorly researched configuration options can result in wide-scale problems when you consider that all of your systems might reflect these same troubles. In this chapter, we will discuss some best practices and methods for handling the deployment of new systems, including the benefits and drawbacks of each.

The Workstation Baseline

A baseline is an established, common starting point from which all workstations begin. Regardless of whether you have a manual or automated process in place, there is normally some process that defines the makeup of your initial workstation installation and configuration. The typical makeup of a workstation baseline is the OS and applications common to all systems. From this point, things might change considerably, with each department (or even individual workstation) getting the additional software packages required for those users to accomplish their jobs. However, the installation of a consistent starting point, automated or not, is the most common first step in the workstation life cycle.

Benefits

Planning and discussing the many options that make up the installation of Windows and its common software is an important step in establishing a common baseline. With planning, a baseline allows for a reliable and consistent starting point, which brings the following benefits: increased reliability, faster deployments, and easier troubleshooting.

Increased Reliability

Through the consistency that comes with the implementation of a baseline configuration, you can establish a firm starting point for all workstations. When all systems start from this proven and tested configuration, there is a decreased likelihood of problems that result from misconfiguration of the system. A well thought out and documented baseline configuration can be directly attributed to a lower number of support calls (and therefore decreased user downtime.)

Increased Deployment Speed

The amount of time and effort needed to get a workstation to a desired state is greatly reduced through the implementation of a workstation baseline configuration. Whether the changes that follow are automated or manual, a repeatable proven process of getting the workstation to a baseline state will reduce overall setup time. In a case in which a duplicated or imaged hard drive is being used in the deployment process, this benefit is more obvious. However, even in cases in which the deployment process is merely a detailed check list, administrators will naturally become proficient in handling this baseline portion of the setup process.

Ease of Troubleshooting

Reproducibility is a key factor in the ability to troubleshoot and solve any computer-related problem. Being able to mimic a problem in a lab environment is often crucial to identifying and testing potential solutions. With a baseline implementation in place, you can more easily troubleshoot problems; you simply apply the baseline installation and add the same delta of additional software packages, and you have a good chance of reproducing a user's problem. Additionally, solutions to problems might be more commonly applied to all workstations. If a problem on one machine is discovered, it is likely the same resolution will apply to other systems as well.

Drawbacks

To be fair, we will also explore the other cons. As with any solution, even the use of a baseline can have negative aspects in certain situations. For example, a problem for one is a problem for all. If you don't get the baseline correct, you risk a problem on all machines. It is for this reason that extensive testing is a necessity in the implementation of a baseline by any means (automated or manual).

In addition, manual repetition can be a problem. When the process of establishing a baseline configuration is not automated, repetition can lead to carelessness. A check list is your best defense against manual misconfigured systems, but an administrator can still grow to feel they have memorized the process and ignore the check list. Forcing administrators to turn in such a check list as a deliverable document as part of the deployment process can result in administrators checking off items blindly simply to produce the required paperwork. Automating the development of your baseline through scripts or a combination of a manual check lists and execution of scripts will also help to ensure consistency in establishing your baseline.

Baseline Components

Planning what should be included in your baseline is obviously a very important step in the process. However, deciding which items to leave in the installation and which not to include is not always an obvious decision.

Table 2.1 shows a simplistic view of what might make up the software requirements for the development, engineering, and graphic art departments at your organization. The baseline configuration to be established here would be all but the last application in each list. Based on this planning document, you might enjoy the benefits of a baseline for these common components and use your application deployment method of choice to add the necessary delta in order to satisfy the requirements of each user or group of users.

Developers	Engineers	Graphic Artists
Microsoft Win2K	Microsoft Win2K	Microsoft Win2K
Microsoft Internet Explorer 6	Microsoft Internet Explorer 6	Microsoft Internet Explorer 6
Microsoft Office XP	Microsoft Office XP	Microsoft Office XP
Adobe Acrobat Reader	Adobe Acrobat Reader	Adobe Acrobat Reader
Virtual Studio .NET	**Visio Technical Edition**	**Adobe Photoshop 7**

Table 2.1: A sample baseline software makeup.

What to Leave In

You can avoid having to deploy and configure multiple instances of the software that is going to everyone (and therefore avoid the inherent risks that come with that process) by simply including the software in a baseline image or unattended scripting process. Even if your deployment process is entirely manual, you should still plan to create a baseline that includes all common software to avoid the need to return to workstations more than once.

What to Leave Out

Due to the popularity of drive imaging or cloning software, many client/server applications that have their own management consoles will provide specific recommendations regarding how they should be handled. These applications might not easily lend themselves to inclusion in a system baseline image or script, and you should handle them via the provided management console. Pay special attention to software with client agents or services such as antivirus software and desktop management tools.

Common Pitfalls to Avoid

As desktop administration becomes more and more transparent to the end user, its complexity can increase for you, the administrator. As a result, there are a growing number of pitfalls to watch out for. We will cover some of these issues in this section.

MSI Source Resiliency

Windows Installer (MSI) introduces a new element to consider as a desktop administrator—installation source availability. When software is installed, Windows Installer will check its installation location for installation files if and when they are needed in the future. If a new feature needs to be installed or if missing files need to be reinstalled, Windows Installer will look to the software's installation source to obtain these files. Thus, when creating images and duplicate drives, be sure that the location where Windows Installer will look is available.

When installing MSI setups from a CD-ROM, mapped drive, or UNC path, you must consider that this path is the one that will be checked by Windows Installer. Do you want all systems looking to the same network share for source files? Did you install Microsoft Office from the CD-ROM when you built your image? Failure to plan for a resilient MSI source might result in user prompts for setup files that will likely result in an increase in support calls.

For an overview of Windows Installer technology and recommendation for how to deal with issues unique to Windows Installer, see *The Definitive Guide to Windows Installer Technology for System Administrators* by Darwin Sanoy and Jeremy Moskowitz available from a link at http://www.realtimepublishers.com.

Non-Essential Data

Another pitfall to watch out for: Avoid inclusion of user- and machine-specific data in your baseline. Although this issue is more of a challenge when you're imaging or duplicating drives, be aware that your user profile might contain information that need not be deployed to all systems. Take a good look around by searching the file system (pay special attention to profile directories and temporary folders) and the Windows registry for your username. The Windows built-in search function, which Figure 2.1 shows, allows you to look for text within any file in the specified path. By specifying all files on all drives and entering your username as the text to search for, you might uncover some files that you do not want sitting on every new machine you roll out. In some environments you might want to avoid including the company name or network name in the baseline image.

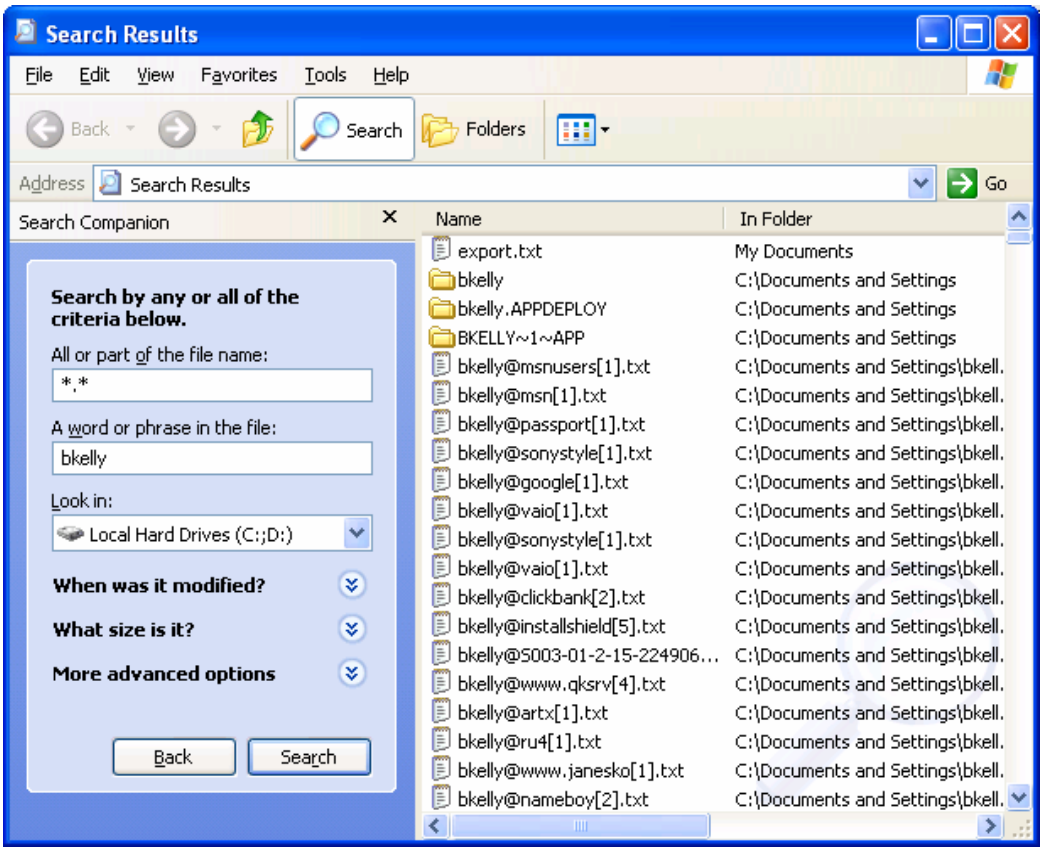

Figure 2.1: Windows' search view.

Classified or Company Proprietary Information

Accidentally including classified or company proprietary information in your baseline might be more than undesirable—doing so could get you in big trouble! If during the development of your image you have opened or accessed classified or company proprietary information, remember to clean up temporary files and cached data before finalizing an image or even before walking away from a manual setup. Though you might have application-specific areas to focus your search for such information, a good start will be to empty the Recycle Bin, delete cached Internet files and temporary folders, and remove unnecessary profiles.

Globally Unique Identifiers

Globally unique identifiers (GUIDs) are being used more and more these days. Be aware that if you include them in your image, they will no longer be globally unique! Client/server applications (client agents or services that communicate with a server console or management system) commonly use GUIDs to identify their client-side systems (such is particularly true for antivirus and desktop management client software). These unique identifiers are often established during installation and might be used to uniquely identify their clients on the network. Have a look at the documentation for any client/server software to ensure that GUIDs will not be a problem for your baseline.

Including Microsoft's SMS client in an image is another common way that a GUID might be inadvertently introduced. SMS assigns a GUID to a client system upon installation, and it is this number that identifies the computer to SMS. This situation can cause a lot of problems for SMS (software installing on the wrong computer, inventory information merged with other clients, and only the last computer to use the GUID will be listed in the management console). One way to end up with the same GUID assigned to all clients is to include the following registry entry in your image: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\SMS\Client\Configuration\Client Properties\SMS Unique Identifier. In addition, there are two files in the Windows System directory that can result in duplicate GUIDs: SMSuid.dat and SMScfg.ini.

The best way to avoid duplicate GUIDs with SMS is to not include the SMS client software in your baseline. SMS and most other client software agents provide their own method of deployment. Alternatively, you can make use of whatever application deployment process you have in place to install SMS as you would any other item that is not part of your baseline. This particular problem is only applicable to imaging and drive-duplication deployment methods. If you're using a manual or scripted process to roll out your baseline, you might be able to automate the installation of the client software so that it might be installed (and generate its own GUID) as designed.

> ☞ Non-unique GUIDs are such a common problem that Microsoft provides newuid.exe, a tool to help remove duplicate GUIDs after they occur. You can read more about dealing with duplicate GUIDs in SMS in the Microsoft article "Managing Duplicate Microsoft Systems Management Server Unique Identifiers."

Profiles

In addition to the fact that leaving user profiles in your baseline takes up unnecessary space, failing to do so might result in personal files on every machine. To avoid this pitfall, on the baseline systems, right-click My Computer, select Properties (or launch the System applet from Control Panel), and use the User Profiles option to delete user profiles from the computer, as Figure 2.2 illustrates.

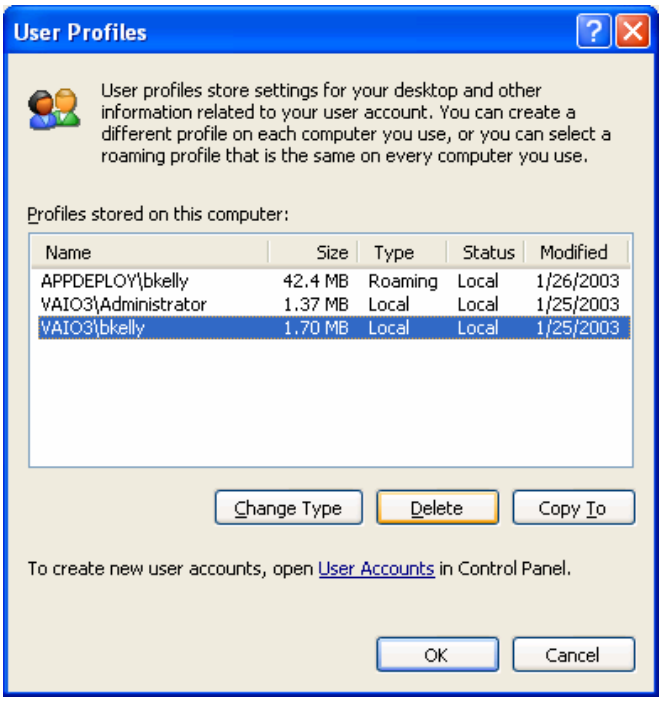

Figure 2.2: User profile dialog box of the Windows System applet.

> Simply deleting the user profile directory (C:\Documents and Settings\<userid> for Win2K and later and C:\WinNT\Profiles\<userid> for Windows NT) does not effectively remove all traces of the user profile.

To automate the process, use the DelProf.exe utility from the Win2K resource kit. However, this task is usually one that you perform manually before finalizing the creation of a duplicate disk or image. Therefore, using the previously mentioned method (illustrated in Figure 2.2) should work fine for most situations and will allow Windows to properly account for the profile and all references to it.

Security Identification Numbers

Users, groups, and computers are referred to by the security system of NT and later using their security identification numbers (SIDs). As Figure 2.3 shows, Windows does not see me as *bkelly*, but as S-1-5-21-592864255-1439916374-227697207-1031. It is important to the Windows security system that these user SIDs are unique so that Windows does not confuse one account or system with another.

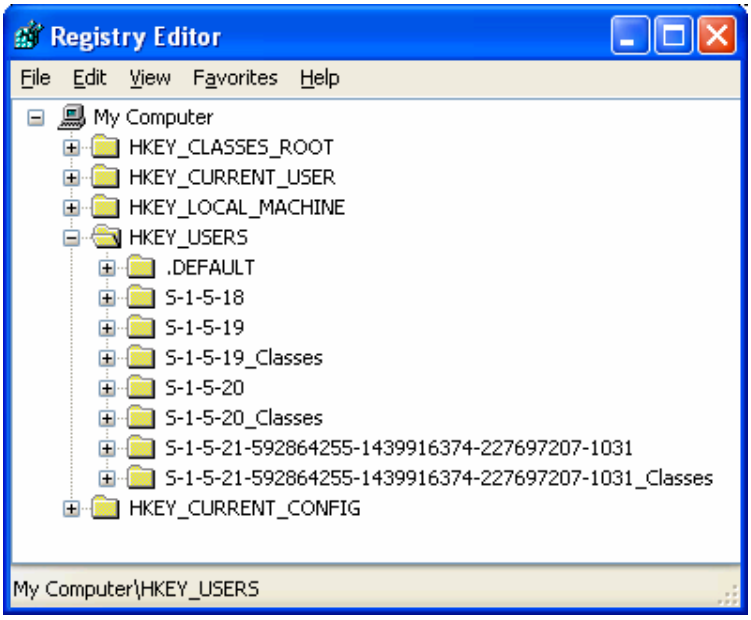

Figure 2.3: Registry editor view of HKEY_USERS.

When a machine is imaged, so are the unique SIDs for the computer and its local accounts. For most environments, this problem isn't as serious as you might have been lead to believe. When a workstation joins a domain, a domain SID is generated for the computer, which results in a unique SID for that workstation. Where this can be a problem is with any computers that do not join a domain (workgroup configurations). When you have two or more computers with the same computer SID, local user and group accounts created on these different computers will generate the same SIDs as they are created. The first account created on one computer and the first account created on another computer (with the same computer SID) will each be assigned the same user SID. In this scenario, both users will have the same security credentials and will therefore be able to access each other's data, even when it is restricted with NTFS file permissions. There are a number of both third-party and Microsoft tools available to address the issue of duplicate SIDs, as the following sections show.

Sysprep

Microsoft Sysprep once did little more than generate a new SID for a system; the latest version of the tool does quite a bit more. The answer file used by Sysprep (Sysprep.inf) provides much of the functionality used to perform an unattended installation, including naming the system, identifying drivers, and setting the local administrator password. For Win2K and later systems, Sysprep can also instruct the system to rebuild its Plug-and-Play (PnP) driver database. I provide more details about Sysprep later in this chapter.

Ghost Walker

Ghost Walker is provided with Symantec's Norton Ghost tool suite to "walk" through a system and replace duplicate SIDs with newly generated SIDs to ensure uniqueness. Ghost Walker also changes the SID for all user profiles on the computer to a unique, randomly generated value. Because both the imaging utility (Ghost) and Ghost Walker run in DOS, changing the SID with Ghost Walker does not require an additional restart as most similar tools do.

SIDgen

SIDgen from Altiris provides a means of changing SIDs in the registry and the file system. SIDgen works by backing up the registry, then generating a unique SID for the computer. It searches all registry hives and replaces appropriate entries with the newly generated SID. SIDgen then replaces SIDs found in the file system (NTFS permissions). You run SIDgen from within Windows, and though it is automated, it does require a reboot when complete.

SIDchanger

SIDchanger from PowerQuest is yet another tool that comes with PowerQuest DeployCenter. SIDchanger supports only NT; PowerQuest recommends using Microsoft's Sysprep for Win2K and later systems.

ImageCast SID Creator

ImageCast from Phoenix Technologies offers SID Creator as its tool to automatically assign a unique SID on each NT and Win2K workstation after imaging. However, the company recommends using Sysprep for this task.

NewSID

Sysinternals' NewSID is a popular free tool for dealing with SID duplication problems. The most recent release includes support for Windows XP and Windows Server 2003, a wizard-style interface, and the ability to specify the SID that you want applied in addition to other enhancements. NewSID also provides full source code.

With so many tools available, it can be a bit confusing to determine which is best for you. If you are not joining a domain or have local user accounts in your baseline, vendor-provided tools such as SIDgen and Ghost Walker provide more thorough scans of the computer, including the file system. However, for most, Sysprep is more than sufficient and is a very powerful tool in the deployment of new systems.

Initial Build Size

Keep your image as small as possible so that it takes up less space (particularly when you need it to fit on removable media). Most tools allow you to span media, but it is certainly more manageable if you can keep your image on one CD-ROM or DVD. With a simple baseline image containing little more than the OS, a compressed image might fit on a single CD-ROM. Although a single CD-ROM image is ideal, the compression you will be able to attain will vary and a single CD-ROM might prove a difficult goal.

Another reason to keep your baseline image small is to increase the speed with which you can apply the image. A smaller image will obviously take less time to process than a larger one. This processing time is of particular concern when it comes to network downloads, as bandwidth limitations will be a factor for a longer period of time. The following list explores ways to keep your initial build size small.

- Reduce Windows File Protection cache—The default size of the Windows File Protection (WFP) cache is 400MB. The same files will be protected by WFP regardless of the cache size, so you can reduce the cache size to decrease the size of your initial build. The downside to doing so is that you might be prompted for an installation CD-ROM or network shared installation files in order for WFP to do its job restoring replaced or removed files.

> 🕮 For information about WFP and controlling the size of the cache, see the Microsoft article "Description of the Windows File Protection Feature."

- Turn off hibernation support—The hibernation file, Hiberfil.sys, is roughly the size of the amount of RAM on the local system, which can be hundreds of megabytes that you need not include in an image. To be rid of the hibernation file, turn off hibernation mode support. To do so, in the Display Control Panel applet, click Power in the Monitor Power area of the Screen Saver tab. Next, select Never from the drop-down list for the *System hibernates* option, and press OK. If you really want hibernation support, consider automating the setting after the system is imaged. Listing 2.1 shows an KiXtart script that activates Windows hibernation support.

```
Run "%Comspec% /c RunDLL32.EXE shell32.dll,Control_RunDLL ups.cpl,2"
Do Sleep 1 Until SetFocus ("Power Options Properties") = 0
$rc = SendKeys("~H")
$rc = SendKeys ("{TAB 4}")
$rc = SendKeys("{RIGHT 4}")
$rc = SendKeys("~H")
$rc = SendKeys("{ENTER}")
```

Listing 2.1: KiXtart script to activate Windows hibernation support.

- Empty the Recycle Bin—By default, the Windows Recycle Bin is set to 10 percent of your hard disk. If you are creating an image, the Recycle Bin certainly should be nowhere near full, but ensure that it is empty to save as much space as possible.
- Clear the event logs—By default, the event logs can be as large as 512KB. In addition to eliminating wasted space, clearing the event logs before you create your baseline image will avoid confusion that arises when events that were inadvertently deployed to systems where the events didn't take place show up in the event logs.
- Delete user profiles—As discussed earlier, deleting user profiles just makes good sense. They can potentially be quite large, but regardless of the size, it is unprofessional to include your own documents, settings, Internet cache, and who know what else in a baseline image to be installed across the network.
- Empty temporary directories—Check the local admin temp directory and the system temp directory for files that might have been generated and not cleaned up during application installations or other setup operations. Use the NT and later built-in utility for cleaning up your hard drive by typing

  ```
  cleanmgr.exe
  ```

  from the Start menu's Run dialog box or at a command prompt. The Disk Cleanup tool will scan the specified drive for temporary files and facilitate their deletion (see Figure 2.4).

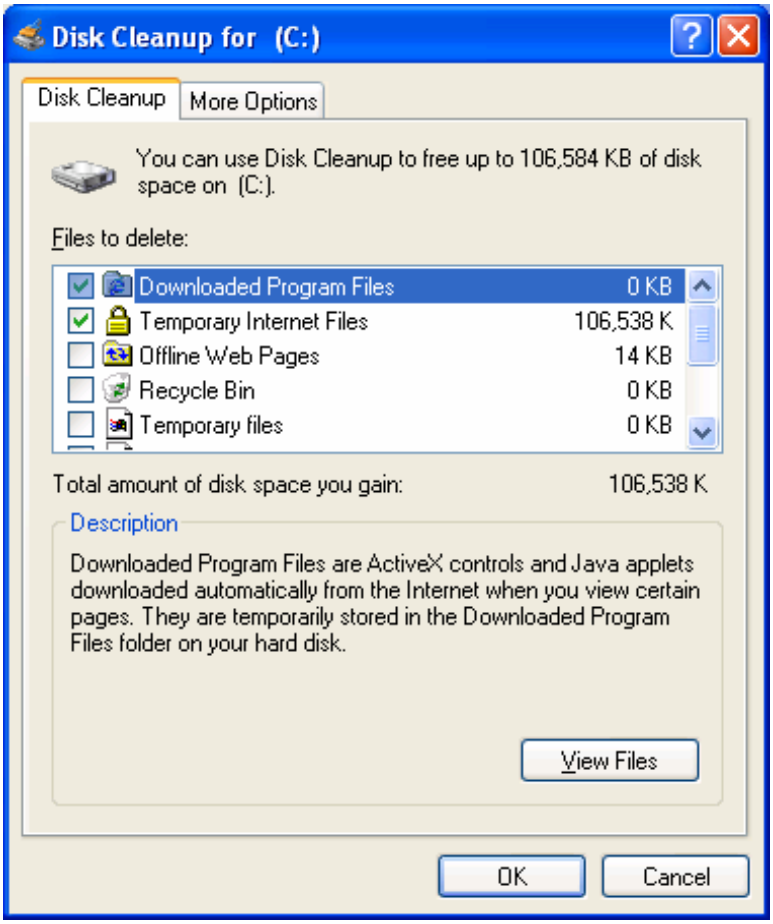

Figure 2.4: The Windows Disk Cleanup utility.

Staging a Deployment

It is a common and effective process to establish a staging area for the installation and initial configuration of new systems. Often an area is set aside to power up computers, apply an image via a local network switch or CD-ROM, and pile them up for distribution.

Benefits

As with many OS deployment considerations, the benefits of putting together a staging area for a deployment will depend on your organization. The following sections explore a few of the common benefits to such a process.

Less Time Spent at the Users' Desks

Having a staging area for your deployment will mean less time spent at users' desks. When you ask users to stand aside while you give them their new (or at least different) computer system, the less time you take the better. The faster you are, the faster you can be on to the next system and the faster the user can be back in business. In the end, the users are your customers, and speed is something they will be impressed with and will result in greater productivity.

Controlled Network Environment

When a network installation (imaged or unattended) is being performed, network speed will be a critical factor in the amount of time each system will take. In the setup of a staging area, you are in a better position to optimize network topology in your favor.

Most imaging software supports multicast network distribution in which one data stream is received by multiple clients simultaneously. Figure 2.5 illustrates a typical data transfer in which individual streams go to all systems.

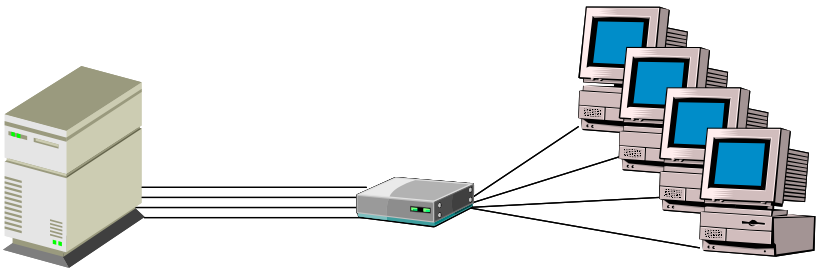

Figure 2.5: Typical data transfer (individual streams to all systems).

As Figure 2.6 shows, multicast is much easier on the network and the server as compared with managing multiple data streams. Although not all network environments support this technology, in a lab environment obtaining and configuring support for multicast should be a much easier implementation. Lack of support is normally in the form of outdated or improperly configured routers, which might require extensive purchases or configuration changes that would make this technology a challenge to implement network-wide. Imaging 10 computers from the same source will prove a considerable advantage over the congestion that comes with 10 individual data streams (all fighting against one another for bandwidth).

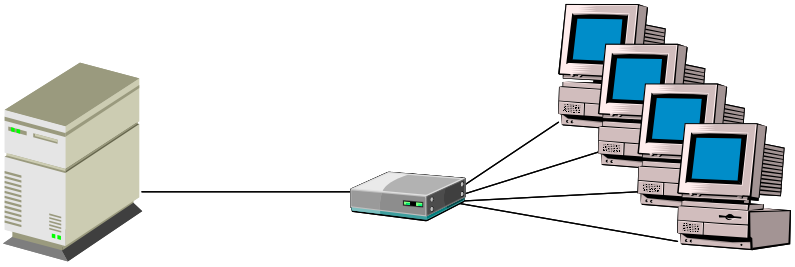

Figure 2.6: Multicast data transfer (single stream to all systems).

Early Identification of Failed Systems

If you purchase 10 computers will they all work? One hundred? One thousand? It is safe to assume that through manufacturing, parts, and shipping (or while you're taking it out of its box) things are going to be broken. It is much more desirable for everyone involved if these systems are identified and addressed in a staging area before you carry them through the building and plugging them in at someone's desk.

Drawbacks

In some environments, a staged deployment simply might not be an option. Your own organization's reasons might be unique; the following sections discuss a couple of reasons that staging might not be a good idea.

Location-Specific Actions

Custom scripts or client/server applications in use on your network might perform automatic actions based on location. In many cases, you can "fool" the system into thinking it is at its final location for staging if such actions are based on the name you give it, computer group membership, or organizational unit (OU) membership. However, when the network subnet is a deciding factor, there may be nothing you can do.

Slow Delivery in a Changing Environment

In a deployment staging area, workstations might sit unused as active computers receive network-wide changes. If you control the rate at which new machines are prepared for deployment, this kind of problem might be minimized. Depending upon your deployment mechanisms, the systems might simply update themselves when connected to the network. In this case, the issue may be less critical. However, the time of the deployment might still be a factor while you wait for the new updates to be applied.

Deployment Methods

There are several deployment methods available; which one is best for you will depend greatly on your budget, size of the deployment, and the skill sets of the deployment team. Much of what we have covered applies to the more common mass deployment methods of imaging and drive duplication. In the following sections, I will describe the common methods used and summarize the benefits and drawbacks of each.

Manual Installation

For those with only a handful of systems to set up, a manual installation is still a common way to go. However, more and more organizations have implemented more robust means of deploying systems that are faster and more consistent even when considering very few workstations. Every method has its up side, and manual installations are no exception:

- GUID uniqueness—Accidental inclusion of GUIDs or other machine-specific items need not be a concern when performing a manual installation. After all, this was how the vendor intended the deployment to be performed.
- Customize to individual needs—Also a negative aspect of manual installations, the ability to customize the installation on the fly is easiest when a manual installation is being performed. If the user has a peripheral that you did not plan on or if there is a software package that must be installed prior to those in your baseline process, a manual process will allow for handling these situations with the least difficulty.

- Hardware installations supported as designed—PnP operations, loading of the appropriate hardware abstraction layer (HAL), and legacy driver detection and installation are all designed to take place dynamically during the installation process. Therefore, both manual and unattended installations enjoy the increased success that comes with performing an installation as it was intended to take place. This support can be a significant obstacle for imaging and drive duplication implementations.
- Windows upgrades—When upgrading Windows to a newer version, manual and unattended installations are the only methods that will allow you to take advantage of an in-place upgrade. Testing is naturally very important here, but the primary benefit of an in-place upgrade is that most applications, data, and settings can be preserved. Of course, any existing problems with the system might also be preserved, but if this is not a concern, an upgrade may be the way to go.

There are a few drawbacks to performing installations manually, some of which are quite obvious:

- Inconsistency—A key reason to avoid manual installation of systems, and applications for that matter, is that the process becomes susceptible to human error and inconsistencies. All the benefits of a tested, planned, consistent baseline discussed earlier are at risk when manual installations are being performed.
- Slow installation times—Among the options available, a manual installation is also the slowest way to go. Even when an administrator is very proficient at installing the baseline configuration, nobody can compete with the speed of an unattended script.
- Multiple reboots—You will have to manually perform multiple reboots, particularly when considering the installation of other baseline software outside the installation of Windows itself. Internet Explorer is well known by administrators to be a real challenge for unattended deployment primarily due to its requirement for a reboot along with a second logon by an administrative account. Windows service packs and hotfixes require reboots as well.

Microsoft has instituted a directive in its logo certification program that requires vendors to avoid the need to require a reboot of the computer. Far too many situations have surfaced in the past in which reboots are requested but not really needed. With Win2K (and even more so in Windows XP), there are far fewer events that require a system restart. Those events that still require a restart might present alternative ways of getting the job done, which eliminates the need to reboot. What all this means to us as administrators is that with any luck, we will see far less prompts for a restart today and in the coming years.

There are a number of ways to handle a manual installation aside from relying upon the administrator's familiarity and experience to capturing screen shots of every step. When it comes to a manual installation, strong documentation is about the only "solution" available to you.

Documentation and check lists are critical components of consistent and accurate installations. Although some organizations include detail on every file version, location, and source, others will simply list the applications to be installed and the order of their installation. Too much detail might be a necessity for some and an unnecessary waste of time for others, but lack of documentation spells trouble for all. Your own configuration management process will dictate the level of detail necessary. Keep in mind that staff changes and reassignments affect an undocumented build process dramatically.

☞ Like most technical challenges you will encounter, chances are that many have faced the problem you're having before you—and some of them might have shared the fruits of their labor. A quick search on the Web for installation and post-installation check lists produces a number of great examples to start from:

Checklist for Upgrading to Windows 2000 Professional at http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/windows2000pro/deploy/upgrdmigrate/prochk.asp.

A Windows 2000 Post-Installation Checklist at http://www.winnetmag.com/Articles/Index.cfm?ArticleID=7700

Windows 2000 Post Installation Checklist at http://www.labmice.net/articles/postinstall.htm

Unattended Installations

Unattended installations are an improvement over manual installations; the benefits and drawbacks of this method will be covered in the following section. As with the other methods available, the benefits of an unattended installation might outweigh those of other methods depending upon your organization. The following list provides common benefits to unattended installations:

- Consistency—A scripted installation is a consistent installation. Removing the human element from the equation is certainly a strong benefit of an unattended installation. Even if the same individual is setting up every machine on a network using a check list and comprehensive documentation, some degree of deviation (deliberate or accidental) becomes inevitable over time.
- Speed of installation—An unattended installation might not be as fast as applying an image or duplicating a drive, but it is certainly faster than following a manual check list and addressing each element of the installation manually. In an unattended installation, many of the options normally presented are not even displayed on the screen. When compared with responding to every option as it is displayed, it becomes clear that an unattended installation is far faster than a manual installation.
- Manual input—You can design unattended installations to give administrators the flexibility to let users provide input during the installation process. By leaving out certain information such as the computer name and product ID, a mini-setup can be utilized that prompts for only these items, while the remainder of the installation is automated as you specify.
- Cost—Out of the box, Microsoft provides built-in and well-documented methods for implementing an unattended installation. We will discuss these options in addition to some third-party solutions that aim to improve upon Microsoft's offerings; however, for many, a Microsoft out-of-box solution can be developed at little or no additional cost.

The following list provides the drawbacks of an unattended installation. Every method has its benefits and drawbacks; the correct solution lies in what is important to your organization. Here we will cover some of the drawbacks to unattended installations.

- Speed of Installation—While this was also listed as a benefit, it is still considered a strike against this method when compared to imaging and drive duplication. Depending upon the software to be installed after the operating system, an unattended installation could easily take over an hour to complete. The source files and scripts may be stored on a CD (or set of CDs.) The problem may then become the space limitation of just one CD or hanging around to insert the next CD. Alternatively, such installations are often handled over the network, where limitations may be imposed by network bandwidth. It will also be necessary to limit the number of simultaneous installations take place before network saturation brings the file transfer process to a crawl.
- Potential Complexity—Depending upon what it is you want to accomplish during your unattended installation (application installations, driver installation, reboots, automatic logons, etc.) an unattended installation has the potential to grow into quite a complex process. An out of the box installation where Windows is simply installed and waits for a logon can be surprisingly easy to accomplish. However, the more you attempt to automate, the more complex the entire process will become. The primary problem with a very complex approach is there is more to go wrong and, at the same time, fewer people will understand and be able to troubleshoot the process.
- Multiple reboots—As with manual installations, multiple reboots can be a problem for unattended installations.

Microsoft Solutions

Slow to adopt support for imaging, Microsoft's preferred method of automated installation has always been unattended installations. To support this perspective, Microsoft has made a handful of tools available that work together to help you out. We will briefly cover each of these utilities and processes in the following section.

Sysprep

The Sysprep.inf file is used by the Microsoft System Preparation tool (Sysprep) to answer the questions normally posed in the graphical wizards and prompts presented during the manual installation of Windows. Sysprep also allows for the clearing of the PnP database, inclusion of additional drivers, and much more. Take advantage of this powerful tool when utilizing any of the automated methods covered in this chapter. Sysprep offers a host of other configuration options as the Sysprep.inf file in Listing 2.2 shows.

```
[Unattended]
UnattendMode = FullUnattended
NoWaitAfterTextMode = 1
NoWaitAfterGUIMode = 1
OemSkipEula = yes
InstallFilesPath = "%systemdrive%\Sysprep\i386"
OemPnPDriversPath = "\Sysprep\drivers"
ExtendOemPartition = 0

[GuiUnattended]
AdminPassword = *
TimeZone=10
OemSkipWelcome = 1
OemSkipRegional = 1

[LicenseFilePrintData]
AutoMode = "PerServer"
AutoUsers = "5"

[UserData]
FullName="FullName"
OrgName="PC"
Computername=*

[Networking]
InstallDefaultComponents = Yes
```

Listing 2.2: A sample Sysprep.inf file.

In Listing 2.2, the asterisk (*) after the computername value results in a random computername based on the orgname value (for example, PC3149361729). Additionally, the number 1 represents True for many of the parameters shown in the listing.

Designed for Win2K, a version of Sysprep was made available for NT only to Microsoft Enterprise partners. This version of Sysprep provided limited functionality; namely, SID generation. Many administrators still think of Sysprep in this limited capacity, but it is a very powerful tool for automated deployments of Win2K and Windows XP. You can find Sysprep documentation online at http://www.microsoft.com/windows2000/techinfo/planning/incremental/Sysprep11.asp and http://www.microsoft.com/windows2000/techinfo/reskit/en-us/deploy/dggb_aut_cwrh.asp.

Sysprep can be used to apply an image to dissimilar hardware configurations. Among the items that need not be the same on target computers are video cards, network adapters, audio cards, and mass storage device drivers. However, the system HAL and Win2K kernel (Ntoskrnl.exe) must be identical.

To use Sysprep, extract the Sysprep.exe and setupcl.exe files from the support.cab file located in the support/deployment folder of the Win2K and Windows XP installation CD-ROMs. You can download Sysprep from http://www.microsoft.com/windows2000/downloads/tools/Sysprep. Copy the files to the computer to be imaged and run Sysprep; doing so will configure the computer to launch a mini-setup at the next startup. What is presented by this mini-setup is determined by the contents of the Sysprep.inf file used by Sysprep to prepare the system for imaging (for an example, see Listing 2.2). Once the machine shuts down, create your image or duplicate drives for delivery to other computers.

Unattend.txt

The Unattend.txt file provides answers to the questions typically posed during a manual installation. For Win2K and later, many think only of the new support for Remote Installation Services (RIS), but these OSs still provide full (and even enhanced) support for use of an Unattend.txt file.

> 🕮 You can find Unattend.txt file format and available parameters at http://support.microsoft.com/?kbid=155197.

You need not create the Unattend.txt file from scratch. Along with the Sysprep and RIPrep information files (discussed later), the Setup Manager Wizard is provided to populate this file, as Figure 2.7 shows. The Setup Manager is included with Sysprep in the deploy.cab file on the Win2K and Windows XP installation CD-ROMs in the Support folder.

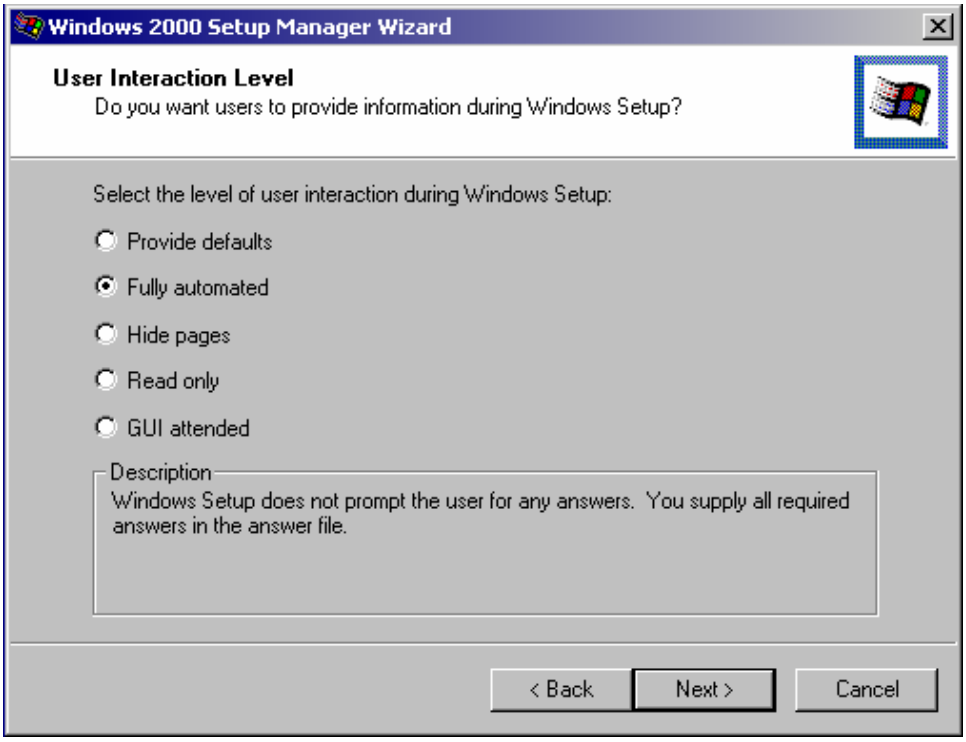

Figure 2.7: The Setup Manager wizard.

> ☞ Windows Product Activation (WPA) might look like a process that must be completed manually. However, Microsoft has implemented WPA only to deter the casual rollout of installations without the licenses to allow it. Microsoft fully supports the automation of this process and provides documentation and recommendations for doing so at http://www.microsoft.com/windowsxp/pro/techinfo/deployment/activation.

OEM

By setting the OEMPreinstall value to Yes in Unattend.txt, you can include additional files in the automated installation process. You can employ the OEM directory structure to add to the source files to include additional drivers and even application installation files. The structure for including these additional files is shown here:

```
<drive>
      /i386
            /$OEM$
```

From within the Setup Manager tool, use the Additional Dirs page to supply your own set of files to be deployed. Doing so will generate the previously shown folder structure for the inclusion of any specified files. For example the files you want placed in the System Drive tree are placed in OEM\$1. Each of the special keys that are displayed on the Additional Dirs page, maps to a special subdirectory of the OEM file structure.

> Windows is hard-coded to look for the OEM folder below the /i386 directory. Because a CD-ROM–based installation copies files to a local temporary directory, the path to OEM might become incorrect. For more information about this potential problem, see the Microsoft article "Cannot Obtain Access to oem Folder During Unattended Setup."

Cmdlines.txt

You can use the Cmdlines.txt file to specify command lines for execution as part of the setup process. Stored in the root of the OEM directory, this file is processed at the end of the mini-setup wizard before saving any settings. To make use of this capability, the OEMPreinstall option must be enabled in the specified unattended answer file.

> For more information about Cmdlines.txt, see the Microsoft article 238955 "HOW TO: Use Cmdlines.txt File During Sysprep.exe Setup Wizard."

Third-Party Solutions

It might seem that Microsoft has deployment tools pretty well covered, but there is always going to be room for improvement. By integrating the native unattended support into a desktop management product or by introducing support for an alternative means of accomplishing the task, you have many alternative solutions available.

- SystemPrep—Sys-Manage's SystemPrep is designed for the automated deployment of Windows as well as additional software and drivers. This tool facilitates the creation of shares and network boot disks and the customization for the installation of Windows releases as early as Windows 95. (Figure 2.8 shows the SystemPrep Configuration Wizard interface.)

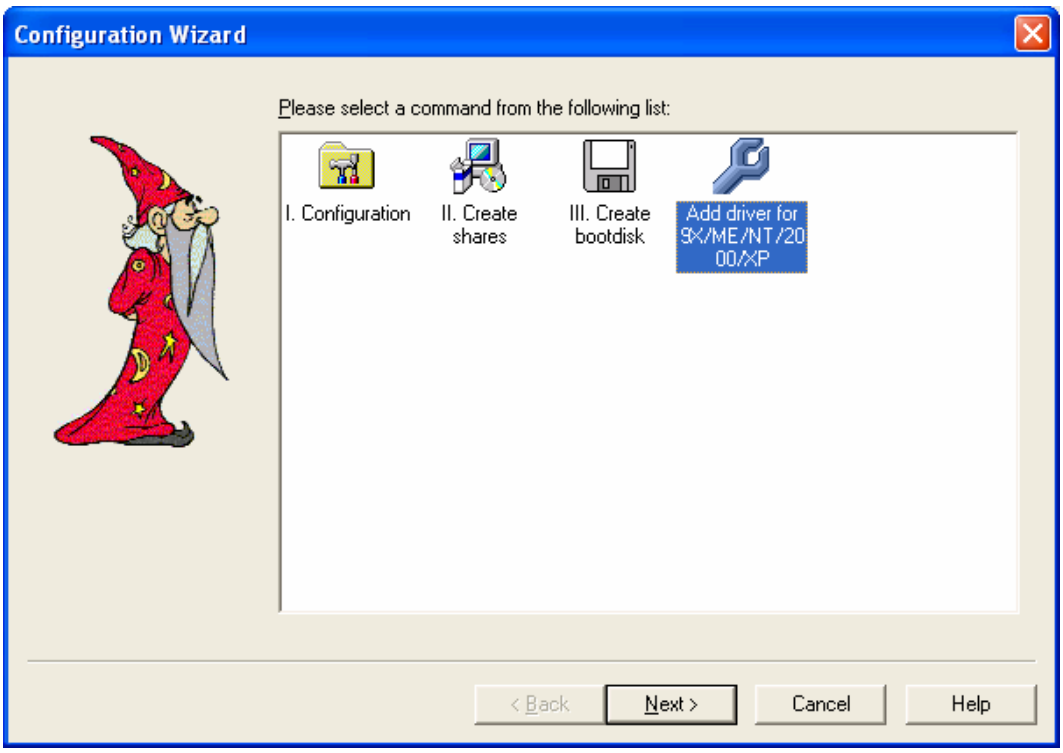

Figure 2.8: The SystemPrep 1.1 Configuration Wizard.

- CapaInstaller PC Creator—CapaSystems' CapaInstaller provides a product to simplify the process of customizing an unattended Win2K or Windows XP installation by asking all the necessary questions and generating a network boot disk to trigger the process. CapaInstaller also features a PC configuration tool, a driver library, a hotfix installation service, a remote reinstall service, and hardware and software inventory.

Drive Imaging

Drive imaging is one of the most popular methods of OS deployment for several reasons, but primarily for its speed and simplicity. But both those benefits assume a successful image has been generated, which can sometimes prove difficult depending on your environment. We will discuss some of the benefits and drawbacks to imaging in the following section.

The benefits of drive imaging include the following items, which have already been discussed:

- Speed of installation—An image can be applied from CD-ROM or a high-speed network connection in just minutes. Considering this image might also include baseline software applications and configurations, speed is a major benefit when compared to other methods available. Factors include the speed of the target system, the method used to provide the image to the system (CD-ROM, network, local partition) and the size of the image itself.
- Consistency—Consistency is a key benefit enjoyed through the imaging process. At least until the next reboot, all computers will be binarily identical at the time the image is applied. When a well-tested baseline is in use, you have a very solid foundation for your new system.

Although a very common means of deploying a large number of systems, imaging has its drawbacks:

- Upgrade is not an option—Imaging constitutes a complete replacement of drive data; therefore, utilizing imaging to perform an in-place update is not possible. One can naturally replace the OS with a newer version, but doing so does not constitute an upgrade in the traditional sense, in which applications and settings are retained or migrated for use on the updated OS.
- All user data and settings are lost—To retain user data and settings, you must arrange to back up this information prior to the installation, then restore it after the installation. Many of the imaging products have addressed this shortcoming of imaging by providing tools to automate this process, even going so far as to incorporate it into a single automated deployment event—backup user data and settings, apply new image, apply user data and settings. Even with such support, added configuration and planning must be applied to the deployment process.

To address this major drawback of imaging (and duplication as well), many of the imaging products available either include or have partnered with other companies with products designed to facilitate the migration of user data and settings between systems. In many cases, these tools are also able to move between versions of Windows and some cases even support the migration of application-specific information and data to newer versions of the application.

In addition, migration of user data between systems becomes increasingly less important (if not completely unnecessary) if folder redirection is used properly. You can perform folder redirection using ScriptLogic, Group Policies, and/or manual registry edits.

- HAL incompatibilities—With the release of Win2K, the number of HAL incompatibilities has decreased significantly to just one key potential incompatibility: Advanced Configuration and Power Interface (ACPI) and non-ACPI. When an incompatibility is present, the lowest common denominator prevails. Thus, a non-ACPI image should (theoretically) work on any ACPI system (but you would lose its ACPI support in the process). In practice, though, you may find it is not so easy. Trial and error dictates the number of images you require to support all systems in your organization. Despite dissimilar system HALs, you might be able to create an image on one machine that applies to several other machines, but an image created of a newer machine is less likely to work on older ones. Here especially, the fewer hardware differences in your environment, the easier your life will be.

NT SCSI and IDE drive images present yet another incompatibility that has thankfully been removed from the picture with the release of Sysprep 1.1.

- Multi-image management—Dealing with multiple images is a necessary evil for many organizations. The desire for multiple baselines and HAL incompatibilities are the core reasons that the number of images to be managed may quickly increase. You might find a need to update images on a regular basis as new applications and updates are added to the deployed baseline. If your desktop management solution is designed to dynamically update the older baseline image after it is applied to a new system, this drawback might be less of a concern. However, it is very likely that there will come a time when the existing network baseline and your original deployment image differ to such a degree that updating the image becomes necessary. When image updates become necessary, the desire for a small number of images becomes evident.

Available Solutions

There are a fairly small number of vendors providing imaging software on the market today. Many of these solutions are briefly discussed in the following section.

> ☞ An independent review of the imaging products available was performed in 2000, which included a comparative matrix of features in several categories. You can find this competitive matrix at http://www.appdeploy.com/comparisons/imaging.

- Remote Installation Services (RIS)—RIS looks more like an unattended installation when you watch it in action, but it is actually a file-based image. Like all of the other imaging solutions available, it can access the network via a network boot disk or PXE (a hardware-based network boot). RIS requires Active Directory (AD) and it supports only Win2K and Windows XP. In addition, it does not support multicasting, which might eliminate it as an option for many enterprise networks planning for large-scale deployments. RIPrep, which Figure 2.9 shows, is a utility provided for the preparation and uploading of images for deployment via RIS.

> 🕮 You can find more information about RIS from the following resources:
>
> RIS installation and configuration check list
> http://www.microsoft.com/windows2000/en/server/help/sag_RIS_Install_Checklist.htm
>
> Technical Guide to Remote Installation Services
> http://www.microsoft.com/technet/prodtechnol/windows2000serv/deploy/depopt/remosad.asp

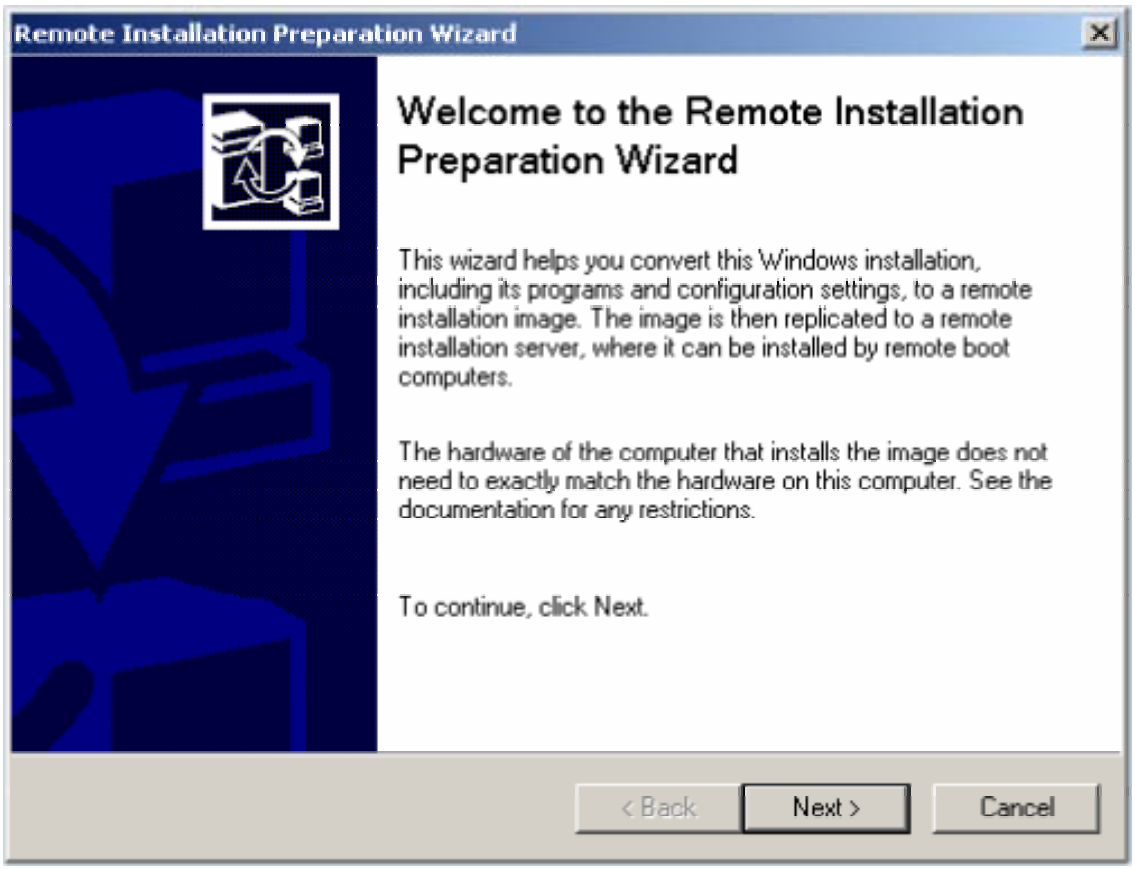

Figure 2.9: The Remote Installation Preparation Wizard (RIPrep.exe).

> ☞ You can customize the text menus presented by the RIS installation process. You can edit the menus or create them with a simple text editor such as Notepad using the OSCML file format. You can find information regarding the modification of these menus as well as its file formatting at http://www.microsoft.com/technet/prodtechnol/windows2000serv/reskit/distsys/part5/dsgappg.asp.

- Ghost—In the imaging market, Symantec Ghost is certainly the most well-known product. In fact, the term *ghosting* has become synonymous with imaging. Ghost provides the ability to image directly to CD-ROM or other removable media. You can use a recordable CD-ROM drive as a place to write an image file to at the time the imaging of a drive is being performed. In addition, Ghost provides USB and LTP peer-to-peer imaging support, offers remote deployment of client management service, and is very fast when compared with other imaging solutions.
- DeployCenter—Formerly Drive Image Pro, PowerQuest's DeployCenter is another very popular imaging tool with plenty of strong features, including PXE and multicast support, image direct to removable media (including CD-R), and a Web-based management console.

- RapiDeploy—Altiris has changed the name of its imaging product from ImageBlaster to RapiDeploy, and the tool is now available as a component of the company's desktop management suite, Altiris Deployment Solution (formerly eXpress.) Although the name changes can be confusing, the product is very intuitive and offers several great features, including complete remote deployment capabilities via its Web-based management console, the ability to create an image of a hard drive and simultaneously distribute that image to other machines at the same time the image file is being generated, remote deployment of client management service support, and direct editing of compressed image files functionality.
- ImageCast—The last release of Phoenix's ImageCast was 4.6.1; however, it is still available in enterprise and manufacturer (MFG) editions. ImageCast's Post Configuration Injector is used to customize each workstation with its individual settings (such as the user name, computer name, and IP address). This tool offers a 100 percent scriptable interface as well as the ability to generate restore CD-ROMs with RestoreBuilder.
- TrueImage—Acronis is a newcomer to the market; its TrueImage product is geared more toward the individual user than an enterprise deployment. However, it is worth mentioning that TrueImage provides the ability to create an image of a machine from within Windows (including the system partition), as Figure 2.10 shows.

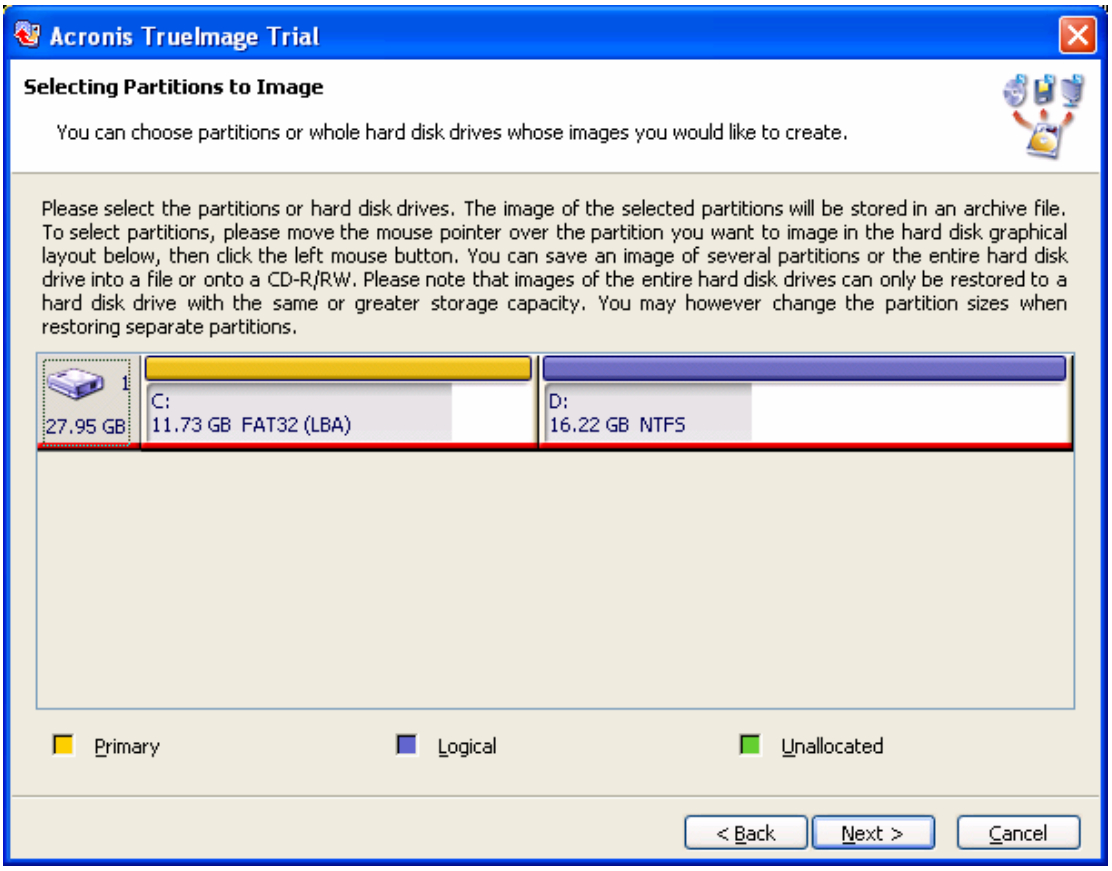

Figure 2.10: The TrueImage partition selection dialog box.

- Auto-Deploy—The main purpose of Rembo Auto-Deploy is to deploy an OS on client computers by replicating a reference system. The deployment process is driven from a central database containing unique parameters for each PC (including the rules that decide which images and software are applied to each computer). The deployment process can be fully automated (without interaction on the client computer), but RAD can also let the final user choose the OS and the applications to deploy. Report logs are then sent to the central console to help the IT administrator control the deployment.

Which Is Right For You?

In many cases, the tool you will use is the one that you are most familiar with. For the most part, imaging software products provide the same feature set. I've listed many of the key differences and standout features, but if you are looking to implement a new solution, do your homework and be sure to get the features that are important to you. Take a look at the features available, create a requirements document stressing what is important to you, and the right choice for your organization will become much more clear.

We will discuss application deployment tools in Chapter 4.

Drive Duplication

Drive duplication is the process of copying hard drives from one to another. This method is a hardware-based solution. The benefits of drive duplication as opposed to the other methods available will depend greatly upon your deployment process. If you want to replace the drive of existing machines with larger ones, drive duplication will likely become a more desirable solution when compared with imaging. The following list provides common benefits of drive duplication:

- Ideal for pre-staging scenarios—There might already be a step in your deployment process that involves opening the case for the installation or removal of components.

Some environments remove floppy drives or modems for security reasons. Others might need to install zip drives or security devices. Even when buying thousands of machines, it is often not cost effective to order the machines customized by the manufacturer—even ordering that a part be removed from a system will likely result in additional costs because it does not fall in line with the manufacturer's process.

- Speed of deployment—Many duplication systems allow you copy from one drive to more than one destination drive at the same time. At faster than 2.3GB per minute, this solution is the fastest available. Add to this the fact that you can simultaneously image as many as 16 drives, and duplication becomes a clear winner where speed is an issue.
- Network bandwidth requirements—Unlike software imaging, and even unattended installations, drive duplication happens off the network, thus not consuming network bandwidth.

- OS independent—With the drives being physically duplicated, the contents of the drives are irrelevant. Therefore, there is no limit to the OSs you can duplicate—any version of Windows, Linux, UNIX, Mac, or any other OS.

All the negative aspects of drive imaging apply to drive duplication. In addition to these drawbacks, drive duplication requires you to open each machine—some systems lend themselves to drive replacement better than others.

Available Solutions

There are several companies providing duplication equipment. We will briefly discuss some options as well as factors you might want to consider when making your decision.

Features to look for in a drive duplication solution include support for IDE and SCSI, the ability to duplicate a large number of drives at one time, and the ability to duplicate dissimilar hard drives. Easy loading and unloading of drives and, of course, speed are also key factors. Some duplicators include diagnostics and utilities, and if you are cloning very large drives, you might want to verify support before purchase (particularly those over 137GB). Some of the larger manufacturers include ImageMASSter, Logicube, Greystone Data Systems, Corporate Systems Center, and Wytron Technology.

OS Deployment Best Practices

OS deployment is a function performed by many desktop, systems, and network administrators. Thus, a number of common sense guidelines are available, many of which were learned the hard way—through experience.

Workstation Naming Conventions

Definitely give some thought to the make up of your workstation names. For identification visually and for use by scripts, a well-engineered naming convention should be a major consideration when starting up a new network or replacing a significant number of systems. This decision is one that you will have to live with for some time. Depending on your environment, different identifying data will make sense. Table 2.2 provides some factors to consider.

Factor	Consideration
Multiple sites	Include a site code or name. If you are using a systems management system such as SMS, you might already have designated site codes in use. See if a site code or identifier is in use on the network before generating new ones.
Multiple buildings	A site might include more than one building. In this case, including something to identify the building number might be a good idea.
Multiple floors	If your network is dispersed between floors or might be in the future, including the floor can be of great help. Does the room number identify the floor already? If so, the floor number might be an unnecessary element.
Multiple wings	Which wing a system is in is also something often identified in a room number, but if it is not and it will help people locate a machine by its name, you can include the wing location as a valuable bit of information.
Multiple rooms	If physically locating a machine is of any interest to you, the room number is likely the most valuable information to include in a workstation name.
More than one computer per room	When there are two or more computers in the same room, and the room number is being used as identifying data for a computer name, you will need to come up with a consistent way of identifying computers within the same room. If there are two, perhaps the one closest to the door is A and the one further away is B. If there are several, you could identify them from distance to the main entrance or perhaps clockwise as you stand in the doorway.
Primary function	If a computer is a shared resource for Internet access or to connect to a certain network, the system's primary function can be useful identifying data to include in the computer name.
Primary user	If computers are assigned to individuals and not rooms, include a UserID or initials. When using initials, keep in mind that the likelihood of duplicate instances will increase along with the number of users. Also keep in mind, people move and company turnover rates might fluctuate. Try to avoid a situation in which you constantly have to change computer names.
Asset tag number	If you have an asset management system in place, using just the asset tag code might suffice. This decision would weigh greatly on the availability and accuracy of your asset management database.
OS	If you have multiple OSs on the network and this information is vital, including an identifier for the OS might make sense. Remember that the OS can change and there are other (possibly better) means of determining and/or documenting the OS for the machine.

Table 2.2: Naming convention factors and the reason to consider them.

A structured computer name can not only help you identify machines visually, but it can also help scripts and other management systems identify machines. For example, if you want to perform an action on all computers at a specific site, and the site is identified as part of your computer name, you can use the computer name as a condition within a script. Several commercial products allow for computer names to be utilized in identifying systems by accepting a computer name as a parameter and offering wildcard support. For example, ScriptLogic's Validation Logic offers the ability to identify which computers should perform a specified action on computer names. Microsoft Operations Manager (MOM) bases the remote deployment of its management agent on computer name. And, when using Deployment Solution from Altiris, you can specify events (such as software deployment) on the computer name. These are just a few examples to help illustrate the value of a meaningful naming convention; check your management software for such support. Even if your software only supports sorting systems by name, a structured naming convention can be of value.

Some example computer name structures include:

- S01B2F1G210A—Site 1, Building 2, Floor 1, G Wing, Room 210, the first or only computer in the room
- ENG-KIOSK03—Engineering group common area workstation number 3
- WNT-KKELLY—An NT workstation that belongs to Kayla Kelly
- PC1032-327G—Asset tag number PC1032 and room number 327G.

Don't include everything—too long and complicated a name might defeat the purpose. You might want to be able to know where the machine is, what its function is, or whom it belongs to. However, computer names are limited to 14 characters. Whatever the decision, try to think beyond your current network configuration. The likelihood that your network will grow in the future is very strong.

Win2K DNS supports the underscore (_) character as a valid DNS character, but other DNS solutions might not, so to avoid possible problems, Win2K Setup recommends underscores be changed to dashes. As a result, it is strongly recommended that you abstain from using underscores in computer names—you might never encounter a problem as a result of using one, but if and when you do, it might not be clear that the underscore is the problem. It is best to simply not use them from the beginning.

System Build Information

Include baseline identification or similar relevant information to the system as part of your rollout process. Best implemented by script, when you include information in the registry for later use, you will be thankful step later.

Image Identification

Perhaps you are lucky enough to have an image that works on all hardware platforms that you need to support. This image is still likely to be the first of many, and a naming convention will prove helpful to distinguish images from one another. If you have different baseline images to track, image identification fits well into an image naming convention. The simplest method is a versioning system. The first image released is version 1, the second is version 2, and so on. If fixes or minor changes have to be made to an image, include a sub-version number (such as 1.2, 1.3, 2.0, and so on). It is also a good practice to include a date on any CD-ROMs that are created, as there may be times during image development at which new and updated images are routinely generated.

Summary

We started this chapter by discussing the concept of establishing a common baseline. We also covered the major deployment methods for rolling out new systems as well as the benefits and drawbacks of each. In the next chapter, we will discuss user profile and user data management.